PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2000-242672

(43)Date of publication of application : 08.09.2000

(51)Int.Cl. G06F 17/50

(21)Application number : 11-045025

(22)Date of filing : 23.02.1999

(71)Applicant : MITSUBISHI ELECTRIC CORP

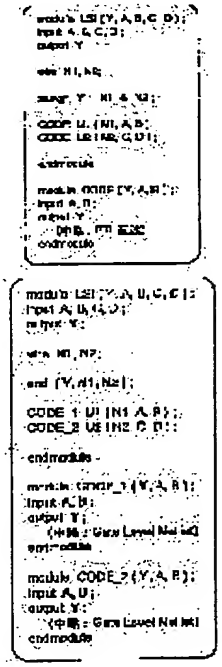
(72)Inventor : WADA KYOJI

(54) DEVICE AND METHOD FOR FORMAL LOGIC VERIFICATION

(57)Abstract:

PROBLEM TO BE SOLVED: To shorten the verification time by lowering the frequency of direct comparison between description lists by comparing an RTL description with one description in a net list regarding a functional block and further comparing descriptions in the net list with one another.

SOLUTION: The formal logic verification device compares the description of a module CODE block in RTL description with the description of a module CODE-1 block in the net list. When the descriptions of those two blocks do not match each other, logic discrepancy is decided. When their descriptions match each, on the other hand, the module CODE-1 block and a module CODE-2 block are compared.



The CODE-1 and CODE-2 blocks are described in the same format. Consequently, they can be compared easier than two blocks in different structures. The formal logic verification can be performed in a short time.

LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

Searching PAJ

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to formal logic verification equipment and the formal logic verification approach, and relates to the formal logic verification equipment and the formal logic verification approach of comparing the contents of the circuit expressed with the contents of the circuit especially expressed with RTL (Register Transfer Level) description, and a gate level netlist (Gate Level Net List).

[0002]

[Description of the Prior Art] Like the design fault of a semiconductor integrated circuit, generally the configuration of the circuit which should be realized is beforehand expressed with RTL description, and the technique of carrying out logic synthesis of the RTL description, and acquiring the netlist of gate level is taken. RTL description expresses the structure of a circuit with the description corresponding to the combinational-logic gate which realizes a specific function, and description of the data transfer between registers, such as F/F and a latch. Moreover, the netlist of gate level expresses the structure of a circuit with the logical expression of a gate component etc.

[0003] Drawing 13 shows an example of the RTL description and the netlist showing the structure of the same circuit. As shown in drawing 13 , in RTL description, only one description is prepared to two or more functional block (SUB U1-U3) which has the same function. To SUB U1-U3, only one description (module SUB (..)) (module SUB (..)) (module SUB (..)) is prepared. Only one description is prepared to SUB U1-U3. In RTL description, the description is used two or more times as a high order hierarchy's (functional block) description. On the other hand, even when two or more functional block which has the same function exists, the description (module SUB1-SUB3) about each of functional block is prepared for a netlist.

[0004] When verifying by formal verification that the logic of RTL description and the logic of a netlist are equivalent (i.e., when verifying that both logic is equivalent by comparing both contents of description), the technique of comparing 1 functional block of RTL description with two or more functional block of a netlist, respectively was taken conventionally (**-** shown in drawing 13).

[0005]
[Problem(s) to be Solved by the Invention] However, since the structures of RTL description and description with a netlist differ, much time amount is needed for comparing the description of functional block and the description of functional block of a netlist format by RTL description. For this reason, long verification time amount was required for the conventional formal verification by which such a comparison is repeated over many times.
[0006] This invention sets it as the 1st purpose to offer the formal logic verification equipment which enables compaction of verification time amount by having been made in order to solve the above technical problems, and lowering the frequency of the direct comparison with RTL description and a netlist. Moreover, this invention sets it as the 2nd purpose to offer the formal logic verification approach which enables compaction of verification time amount by lowering the frequency of the direct comparison with RTL description and a netlist.
[0007]
[Means for Solving the Problem] Invention according to claim 1 is formal logic verification equipment which compares RTL description with the netlist of the gate level obtained by carrying out logic synthesis of said RTL description, and verifies both logic equivalence. One of two or more description about said functional block contained in a netlist when two or more same functional block is contained in a circuit It is characterized by having RTL and a netlist comparison means to compare the RTL description about said functional block, and a netlist comparison means [mutual / each / of two or more description about said functional block contained in a netlist].
[0008] Invention according to claim 2 is formal logic verification equipment according to claim 1, and while having an information generation means corresponding to the block which generates the information corresponding to a block that correspondence with the instance name on the RTL description about the same functional block contained in a circuit and the instance name on a netlist is expressed, said netlist comparison means is characterized by to extract the description about the same functional block out of a netlist based on said information corresponding to a block.
[0009] Invention according to claim 3 is formal logic verification equipment according to claim 1 or 2, and while having a hierarchy generation means to generate the hierarchy of functional block contained in a circuit, based on RTL description, either [at least] said RTL

and netlist comparison means, or said netlist comparison means is characterized by turning the comparison about said functional block to a high order hierarchy, and performing it gradually from the lowest hierarchy.

[0010] Invention according to claim 4 is formal logic verification equipment according to claim 3, and either [at least] said RTL and netlist comparison means, or said netlist comparison means is characterized by excepting compared description from the description about functional block for a comparison by processing of a low order hierarchy, and performing comparison processing.

[0011] Invention according to claim 5 is formal logic verification equipment according to claim 4, either [at least] said RTL and netlist comparison means, or said netlist comparison means is equipped with a check list generation means to generate the check list which records a compared block, and it is characterized by extracting the description excepted from the object of comparison processing with reference to said check list.

[0012] Invention according to claim 6 is claim 1 thru/or formal logic verification equipment of five given in any 1 term. Said RTL and netlist comparison means In advance of said netlist comparison means, the comparison about functional block is performed until RTL description and description in agreement are recognized in a netlist. Said netlist comparison means Description of a netlist is compared for description of the netlist with which coincidence of logic with RTL description was verified by said RTL and netlist comparison means as the first criteria description. When the inequality during description of a netlist has been recognized by said netlist comparison means, said RTL and netlist comparison means The comparison about functional block is again performed until RTL description and description in agreement are recognized out of the description which is not compared [which is included in a netlist]. Said netlist comparison means It is characterized by performing the comparison of description of a netlist again by considering description of the new netlist with which coincidence of logic with RTL description was verified as criteria description.

[0013] Invention according to claim 7 is claim 1 thru/or formal logic verification equipment of five given in any 1 term. Said RTL and netlist comparison means In advance of said netlist comparison means, the comparison about functional block is performed until RTL description and description in agreement are recognized in a netlist. Said netlist comparison means Description of a netlist is compared for description of the netlist with which coincidence of logic with RTL description was verified by said RTL and netlist comparison means as the first

criteria description. When the inequality during description of a netlist has been recognized by the comparison, it is characterized by resuming the comparison for the description which is not compared [which is included in a netlist considering description of the netlist with which coincidence of logic with RTL description is already verified as the first criteria description].

[0014] Invention according to claim 8 is claim 1 thru/or formal logic verification equipment of seven given in any 1 term. Said RTL and netlist comparison means Information is generated as a result of [1st] expressing the comparison result of the RTL description and the netlist in each comparison point included in the description about functional block. Said netlist comparison means Information is generated as a result of [2nd] expressing the comparison result between the netlists in each comparison point included in the description about functional block. And when the inequality of logic has been recognized by said netlist comparison means, it is characterized by having an inequality information generation means to generate the inequality information which expresses the contents of an inequality of description of a netlist, and RTL description based on information said 1st and 2nd results.

[0015] Invention according to claim 9 is the formal logic verification approach of comparing RTL description with the netlist of the gate level obtained by carrying out logic synthesis of said RTL description, and verifying both logic equivalence. When two or more same functional block is contained in a circuit, the RTL description and logic about said functional block until a match is recognized in two or more description about said functional block contained in a netlist The step which compares RTL about the functional block with description of a netlist, After RTL description and the description whose logic corresponds have been recognized in a netlist, description of the netlist as the first criteria description It is characterized by having a step [mutual / each / of two or more description about said functional block contained in a netlist].

[0016] While invention according to claim 10 is the formal logic verification approach according to claim 9 and having the step which generates the hierarchy of functional block contained in a circuit based on RTL description At either [at least] the step which compares said RTL description and description of said netlist, or the step which compares description of said netlist It is characterized by performing the comparison about said functional block gradually towards a high order hierarchy from the lowest hierarchy.

[0017] Invention according to claim 11 is the formal logic verification approach according to claim 10, and is characterized by excepting compared description from the description about

functional block for a comparison by processing of a low order hierarchy, and performing comparison processing at either [at least] the step which compares said RTL description and description of said netlist, or the step which compares description of said netlist.

[0018] Invention according to claim 12 is claim 9 thru/or the formal logic verification approach of 11 given in any 1 term. Until RTL description and the description whose logic corresponds are recognized out of the description which is not compared [which is included in a netlist], after the inequality has been recognized between description of a netlist Description of the step which performs the comparison with RTL description and a netlist again, and the new netlist with which coincidence of logic with RTL description was verified by the processing as the first criteria description It is characterized by performing the step which performs the comparison of description of a netlist again, and **.

[0019] Invention according to claim 13 is claim 9 thru/or formal logic verification equipment of 11 given in any 1 term, and after the inequality has been recognized between description of a netlist, it is characterized by resuming the comparison for the description which is not compared [which is included in a netlist] considering description of the netlist with which coincidence of logic with RTL description is already verified as the first criteria description.

[0020] Invention according to claim 14 being claim 9 thru/or the formal logic verification approach of 13 given in any 1 term, and comparing RTL description with description of a netlist The step which accumulates the comparison result in each comparison point included in the description about functional block, and generates information the 1st result, The step which generates information as a result of [2nd] expressing the comparison result in each comparison point included in the description about functional block, comparing description of a netlist, When the inequality of logic has been recognized between description of a netlist, it is characterized by having the step which generates the inequality information which expresses the contents of an inequality of description of a netlist, and RTL description based on information said 1st and 2nd results.

[0021]

[Embodiment of the Invention] Hereafter, the gestalt of implementation of this invention is explained with reference to a drawing. In addition, the explanation which gives the same sign to the element which is common in each drawing, and overlaps is omitted.

[0022] Gestalt 1. drawing 1 of operation shows the block diagram of the circuit made into the processing object of the formal logic verification equipment of the gestalt 1 of operation of

this invention. The circuit shown in drawing 1 is equipped with LSI10 as a high order hierarchy block. LSI10 is equipped with two CODE(s) 12 and 14 and the AND gate 16 as a low order hierarchy block.

[0023] Drawing 2 shows the RTL description about the circuit shown in drawing 1 . Moreover, drawing 3 shows the netlist of the gate level obtained by carrying out logic synthesis of the RTL description shown in drawing 2 . As shown in drawing 2 , in RTL description, two instance declarations about Block CODE are used corresponding to CODE 12 and 14. Moreover, during RTL description, only one description (module CODE (Y, A, B)) is prepared to those CODE(s) 12 and 14.

[0024] As shown in drawing 3 , each of CODE 12 and 14 is separately described by the netlist of gate level. That is, in a netlist, the description (module CODE_1 (Y, A, B)) corresponding to the block of CODE12 and the description (module CODE_2 (Y, A, B)) corresponding to CODE14 exist.

[0025] The formal logic verification equipment of this operation gestalt creates the information corresponding to a block that the relation between the block information on RTL description and the block information in a netlist is expressed. In case drawing 4 carries out logic synthesis of the RTL description and generates a netlist, it shows the information corresponding to a block that it means that CODE_1 and CODE_2 are generated from CODE of RTL description. Formal logic verification equipment performs the comparison with RTL description and a netlist using the information corresponding to a block.

[0026] Specifically, the formal logic verification equipment of this operation gestalt compares description of a CODE block of RTL description with the description of CODE_1 block in a netlist first based on the information corresponding to the block shown in drawing 4 . Consequently, when description of these two blocks is not in agreement, the judgment of a logic inequality accomplishes. On the other hand, CODE_1 block is compared with CODE_2 block when those description is in agreement next.

[0027] The comparison with CODE_1 block and CODE_2 block is equivalent to the comparison with the CODE block in RTL description, and the CODE_2 block in a netlist. Therefore, according to comparing CODE_1 block with CODE_2 block, the comparison with a CODE block (RTL description) and CODE_2 block (netlist) is omissible.

[0028] Both CODE_1 block and CODE_2 block, it is described according to the format of a netlist, i.e., the same format. For this reason, those comparisons can be easily performed as

compared with the comparison, i.e., the comparison of two blocks with which structures differ, with a CODE block (RTL description) and CODE_2 block (netlist). Therefore, according to the formal logic verification equipment of this operation gestalt, as compared with equipment [functional block of RTL description / functional block / which exists in a netlist / two or more], formal logic verification can be performed in a short time one by one.

[0029] With reference to drawing 5 and drawing 6 , the formal logic verification equipment of the gestalt 2 of operation of this invention is explained with gestalt 2. next drawing 1 thru/or drawing 4 of operation. Drawing 5 shows the 2nd example of the RTL description about the circuit (LSI10) shown in drawing 1 . When are expressed with RTL description as LSI10 shows to drawing 5 and the formal logic verification equipment of this operation gestalt reads the RTL description, it recognizes the layered structure of a circuit. The top hierarchy's functional block is LSI and, specifically, it recognizes that CODE 12 and 14 exists in the low order hierarchy, and the ADD block which is not illustrated by drawing 1 exists further at the lower layer hierarchy.

[0030] The formal logic verification equipment of this operation gestalt generates the hierarchy showing the structure again, after recognizing the layered structure of a circuit. Drawing 6 shows the hierarchy generated corresponding to the RTL description shown in drawing 5 . In drawing 6 , the figure indicated behind a keyword "level" expresses hierarchy level. In this operation gestalt, the top hierarchy's hierarchy level is "0." The hierarchy of a block takes the figure of hierarchy level for falling, and it becomes large.

[0031] In case the formal logic verification equipment of this operation gestalt performs formal logic verification about LSI10, it recognizes that functional block of the lowest hierarchy of LSI10 is ADD based on the hierarchy shown in drawing 6 . And formal logic verification equipment is performed one by one by the technique of the gestalt 1 operation of the comparison about a high order hierarchy's functional block (CODE and LSI), after performing by the technique of the gestalt 1 operation of the comparison about an ADD block and completing the whole of the comparison first, in case the comparison with RTL description and a netlist is performed.

[0032] When the comparison of a low order hierarchy's functional block (for example, ADD) precedes with the comparison of a high order hierarchy's functional block (for example, CODE), and is performed, the part which has already ended logic verification is contained in description about the block in the phase where comparison processing for a high order

hierarchy's functional block is performed. In case the formal logic verification equipment of this operation gestalt performs formal logic verification about each functional block, it excepts the part with which logic verification can already be managed in the comparison with RTL description and a netlist, or the comparison of netlists, and performs it. For this reason, according to the formal logical circuit verification equipment of this operation gestalt, the overlapping useless processing can be eliminated and the comparison with RTL description and a netlist can be performed in a short time.

[0033] With reference to drawing 7 thru/or drawing 9 , the gestalt 3 of operation of this invention is explained with gestalt 3. next drawing 1 thru/or drawing 5 of operation. Drawing 7 shows the 2nd example of the netlist about the circuit (LSI10) shown in drawing 1 , and the netlist obtained by more specifically carrying out logic synthesis of the RTL description shown in drawing 5 .

[0034] The single description (description about CODE) prepared in common to two or more CODE blocks (U1 and U2) and the single description (description about ADD) prepared in common to two or more ADD blocks (SU1, SU2, ..) are included in RTL description shown in drawing 5 . On the other hand, the description about each (CODE_1 and CODE_2) of two or more CODE blocks and the description about each (ADD_11, ADD_12, ADD_21, and ADD_22) of two or more ADD blocks are included in the netlist shown in drawing 7 .

[0035] The formal logic verification equipment of this operation gestalt creates a comparison check list as shown in drawing 8 , when performing the comparison with RTL description and a netlist. The items corresponding to the item, i.e., CODE, and ADD of the description in RTL description are enumerated by the column of the left-hand side in drawing 8 . Moreover, the description about the item of the description in a netlist, CODE_1 [i.e.,], CODE_2, ADD_11, ADD_12, ADD_21, and ADD_22 is included in the column of the right-hand side in drawing 8 . The items corresponding to CODE and ADD are enumerated. In the check list shown in drawing 8 , those items are expressed with the suffix "_1" showing having not completed comparison processing.

[0036] The formal logic verification equipment of this operation gestalt performs the comparison with RTL description and a netlist by the approach of the gestalt 2 operation. That is, a comparison is started from the lowest hierarchy's functional block, and RTL description is compared with a netlist by the approach of omitting the duplicate comparison, on a high order hierarchy. Furthermore, a check is given to the item of the functional block whenever

the comparison processing about each functional block ends the formal logic verification equipment of this operation gestalt.

[0037] Drawing 9 shows the condition of the check list in the phase which all comparisons about an ADD block ended. In drawing 9 , the item of functional block which comparison processing ended is expressed with the suffix "_0" showing completion of processing. In case formal logic verification equipment performs the comparison about a CODE block, with reference to a ** check list, it excepts the description about the ADD block (ADD_11, ADD_12, ADD_21, and ADD_22) which is already comparison ending to drawing 9 , and performs comparison processing to it. According to the above-mentioned processing, memory space and disk capacity required in order to compare RTL description with a netlist are reducible.

[0038] The gestalt 4 of gestalt 4. of operation, next operation of this invention is explained. Like the gestalt 1 of operation thru/or the equipment of 3 mentioned above, the formal logic verification equipment of this operation gestalt compares netlists, when one of description of functional block by RTL description and the description of functional block of a netlist format is compared and both are in agreement. The formal logic verification equipment of this operation gestalt has the description at the point of performing the above-mentioned processing anew only for unsettled description among netlists, when an inequality is detected by the comparison of netlists.

[0039] It explains the case where it is set beforehand that ADD_11 contained in a netlist, ADD_12, ADD_21, and ADD_22 (refer to drawing 7) are hereafter made into the object of comparison processing in the order in the comparison about the lowest hierarchy's functional block, i.e., an ADD block.

[0040] Under the conditions mentioned above, description (drawing 5) of the ADD block which RTL description depends is first compared with the description (drawing 7) of ADD_11 in a netlist in the comparison about an ADD block. Consequently, when both are in agreement next, the description in a netlist, ADD_11 [i.e.,], and ADD_12 are compared.

[0041] If ADD_11 and ADD_12 are in agreement, when being distinguished as a result of the above-mentioned comparison, future processings are advanced like the case of other operation gestalten mentioned above. On the other hand, if it is distinguished that ADD_11 and ADD_12 are inharmonious, the equipment of this operation gestalt compares with description (drawing 5) of the ADD block by RTL description ADD_21 which are a processing object

following ADD_12. Consequently, when both are in agreement, the comparison with ADD_21 and ADD_22 is performed.

[0042] According to the above-mentioned processing, it cannot be concerned with the comparison result about each functional block, but the comparison with RTL description and a netlist can be performed about all functional block contained in LSI10. Therefore, according to the formal logic verification equipment of this operation gestalt, logic verification for the whole circuit can be performed.

[0043] By the way, in the above-mentioned operation gestalt, when an inequality is accepted by the comparison of netlists, although [description of the netlist which is the following candidate for a comparison] compared with RTL description, this invention is not limited to this. That is, when the netlist with which coincidence with RTL description is checked already exists, it is good also as comparing with the netlist the netlist which is the following candidate for a comparison.

[0044] With reference to drawing 10 , the gestalt 5 of operation of this invention is explained with gestalt 5. next drawing 1 thru/or drawing 9 of operation. Drawing 10 is drawing for explaining how the formal logic verification equipment of this operation gestalt generates inequality information, i.e., the information which shows the part RTL description and whose netlist do not correspond.

[0045] In drawing 10 , the term to which REF or REV was given is the term which serves as criteria of comparison processing, respectively, or a term made into the object of comparison processing. Moreover, in drawing 10 , reg100_reg and reg200_reg show the comparison point of it ADD_11 (netlist) and ADD_12 (netlist) for the comparison point [in / in reg100 / an ADD block (RTL description)]. Furthermore, in drawing 10 , "=" and "!=" express that the logic of the term of those both sides is equivalent, and that it is not equivalent, respectively.

[0046] In drawing 10 , a formula (b) expresses that the comparison point reg100 of an ADD block (RTL description) and comparison point reg100_reg of ADD_11 (netlist) are equivalent. Formal logic verification equipment performs the comparison with ADD_11 (netlist) and ADD_12 (netlist), when an ADD block (RTL description) and ADD_11 (netlist) are equivalent next.

[0047] In drawing 10 , a formula (a) expresses that the comparison point reg 100_100 of ADD_11 and comparison point reg200_reg of ADD_12 are not equivalent. In this operation gestalt, formal logic verification equipment extracts the comparison point reg 100_100 of

ADD_11 (netlist), and the point reg100 in agreement out of the comparison point of an ADD block (RTL description) with reference to the comparison result (a formula (b) is included) of an ADD block (RTL description) and ADD_11 (netlist), when such a judgment is made. Formal logic verification equipment generates the inequality information (formula (c)) about comparison point reg200_reg further using the extracted comparison point reg100.

[0048] The formal logic verification equipment of this operation gestalt performs inequality information generation processing mentioned above about the comparison point in all functional block. Therefore, the inequality information on RTL description and a netlist is generable, attaining improvement in the speed of processing by comparing netlists according to the formal logic verification equipment of this operation gestalt.

[0049] With reference to gestalt 6. next drawing 11 , and drawing 12 of operation, the formal logic verification equipment of the gestalt 6 of operation of this invention is explained. Drawing 11 is a flow chart showing a series of processings performed in the formal logic verification equipment of this operation gestalt. According to a series of processings shown in drawing 11 , the gestalt 1 of operation thru/or the function of 5 mentioned above is realizable.

[0050] A series of processings shown in drawing 11 are performed whenever activation of formal logic verification with RTL description and a netlist is required. If the above-mentioned demand arises, as for formal logic verification equipment, processing of step 20 will be performed first.

[0051] At step 20, the source file of the RTL description about the circuit which is the object of processing, and a netlist is read from a database. In addition, the information corresponding to a block as shown in above-mentioned drawing 4 is included in the source file read at this step 20.

[0052] At step 22, while a hierarchy as shown in drawing 6 based on RTL description is generated, based on the hierarchy, the lowest hierarchy's block is recognized out of functional block contained in a circuit. Moreover, at this step 22, functional block which should be made the object of comparison processing is listed based on the information (drawing 4) corresponding to a block.

[0053] At step 24, functional block which should be made the object of comparison processing in this processing cycle is chosen. At this step 24, a hierarchy's low functional block (functional block with the large number of level shown in drawing 6) is given priority to and chosen as a hierarchy's high block (small block of the number of level shown in

drawing 6). The function of the gestalt 2 of operation is realized by the above-mentioned processing.

[0054] At step 26, the comparison with RTL description and a netlist is performed about functional block chosen as a processing object. Comparison processing of this step 26 is efficiently performed using a check list as shown in drawing 8 and drawing 9 , avoiding duplication of the comparison point. Moreover, in processing of this step 26, whenever the processing about each comparison point is completed, a comparison result as shown in drawing 10 is generated.

[0055] At step 28, it is distinguished whether two compared description is in agreement. Consequently, if both are in agreement, when being distinguished, processing of step 30 is performed next. On the other hand, when both were not in agreement and it is distinguished, processing of step 34 is performed next.

[0056] At step 30, it is distinguished whether the block with which it has the same function as the block made into the object of a comparison [in functional block listed by processing of the above-mentioned step 22] in this processing cycle with reference to the information corresponding to a block as shown in drawing 4 , and comparison processing is not yet performed exists. Consequently, if such functional block does not exist, when being distinguished, processing of step 38 is performed next. If functional block which fulfills the above-mentioned conditions exists, when being distinguished on the other hand, processing of step 32 is performed next.

[0057] Description of the netlist determined as the next comparison candidate under the predetermined regulation and description of the netlist format of having been used for the latest comparison processing are compared by step 32. In this step 32, the efficient comparison using a check list (drawing 8 and drawing 9) is performed like the case of the above-mentioned step 26. Consequently, the function of the gestalt 3 of operation is realized. Moreover, at this step 32, the inequality information shown in drawing 10 is generated, performing the comparison during description. Consequently, the function of the gestalt 5 of operation is realized. Termination of processing of this step 32 performs processing of the above-mentioned step 28 again.

[0058] Drawing 12 shows the sequence of the comparison accompanying processing of steps 26-32 mentioned above. The comparison during the description which attaches and expresses ** with step 26 mentioned above in drawing 12 is performed. Moreover, according to the

repeat of steps 28-32 mentioned above, the comparison during the description which attaches and expresses ** or ** in drawing 12 is performed. Thus, according to the procedure mentioned above, the function of the gestalt 1 of operation is realizable.

[0059] At step 34, it is distinguished whether the block with which it has the function same like the case of the above-mentioned step 30 as the block made applicable to a comparison in this processing cycle, and comparison processing is not yet performed exists. When such functional block does not exist, processing of step 38 is performed next. On the other hand, when the above-mentioned functional block exists, processing of step 36 is performed next.

[0060] At step 36, one of the functional block with which comparison processing is not yet performed is considered as the block for a comparison. Termination of processing of this step 36 performs processing after the above-mentioned step 26 again henceforth. According to such procedure, when the inequality of logic is accepted about which functional block in process of formal logic verification, it can verify about all functional block. The function of the gestalt 4 of operation is realized by the above-mentioned processing.

[0061] At step 38, one step of hierarchy of the block for a comparison is raised based on a hierarchy as shown in drawing 6 .

[0062] At step 40, it is distinguished whether the comparison processing about a highest floor layer is completed. Consequently, after processing of a highest floor layer is not yet completed, when being distinguished, processing after step 24 is performed about the hierarchy made a new processing object by processing of the above-mentioned step 38. On the other hand, after processing of the Mogami hierarchy is already completed, when being distinguished, a series of processings shown in drawing 11 are ended.

[0063] Like ****, by performing a series of processings shown in drawing 11 , all of the function of the gestalten 1-5 of operation can be realized, and, according to the formal logic verification equipment of this operation gestalt, the comparison with RTL description and a netlist can be performed efficiently.

[0064]

[Effect of the Invention] Since this invention is constituted as explained above, effectiveness as taken below is done so. According to invention according to claim 1 or 9, the comparison frequency of RTL description and a netlist can be lowered by comparing description of a netlist. For this reason, according to this invention, formal logic verification of a circuit can be performed in a short time.

[0065] According to invention according to claim 2, the relation between the instance name in RTL description and the instance name in a netlist can be easily grasped by referring to the information corresponding to a block. For this reason, according to this invention, the description about the same functional block, i.e., the description which should be compared mutually, can be easily extracted out of description of a netlist.

[0066] According to invention according to claim 3 or 10, while being able to grasp the layered structure of a circuit easily based on a hierarchy, a comparison can be performed from a low order hierarchy using the information. Therefore, according to this invention, description about functional block can be compared efficiently.

[0067] According to invention according to claim 4 or 11, the block compared by processing of a low order hierarchy can be excepted, and comparison processing of a high order hierarchy can be performed. Therefore, according to this invention, formal logic verification of a circuit can be performed efficiently.

[0068] According to invention according to claim 5, functional block [finishing / a comparison] is easily detectable by referring to a check list. For this reason, according to this invention, the block which should be excepted from comparison processing is easily detectable.

[0069] According to invention according to claim 6 or 12, after the inequality of logic has been recognized by the comparison of description of a netlist, the formal logic verification which targets description of a non-compared netlist for RTL description to criteria description anew can be started. Therefore, according to this invention, even when the part of a logic inequality is contained in RTL description and a netlist, formal logic verification can be performed about all functional block.

[0070] According to invention according to claim 7 or 13, after the inequality of logic has been recognized by the comparison of description of a netlist, the formal logic verification for description of a non-compared netlist can be started by considering description of the netlist with which coincidence of logic is already verified as criteria description. Therefore, according to this invention, formal logic verification about all functional block can be performed in a short time.

[0071] According to invention according to claim 8 or 14, it replaces with the comparison with RTL description and description of a netlist, and in spite of comparing description of a

netlist, the inequality information on RTL description and a netlist can be acquired.

CLAIMS

[Claim(s)]

[Claim 1] It is formal logic verification equipment which compares RTL description with the netlist of the gate level obtained by carrying out logic synthesis of said RTL description, and verifies both logic equivalence. One of two or more description about said functional block contained in a netlist when two or more same functional block is contained in a circuit Formal logic verification equipment characterized by having RTL and a netlist comparison means to compare the RTL description about said functional block, and a netlist comparison means [mutual / each / of two or more description about said functional block contained in a netlist].

[Claim 2] While having an information generation means corresponding to the block which generates the information corresponding to a block that correspondence with the instance name on the RTL description about the same functional block contained in a circuit and the instance name on a netlist is expressed, said netlist comparison means is formal logic verification equipment according to claim 1 characterized by to extract the description about the same functional block out of a netlist based on said information corresponding to a block.

[Claim 3] It is formal logic verification equipment according to claim 1 or 2 characterized by turning to a high order hierarchy a comparison concerning said functional block in either [at least] said RTL and netlist comparison means, or said netlist comparison means while having a hierarchy generation means to generate the hierarchy of functional block contained in a circuit, based on RTL description from the lowest hierarchy, and carrying out gradually.

[Claim 4] Either [at least] said RTL and netlist comparison means, or said netlist comparison means is formal logic verification equipment according to claim 3 characterized by excepting compared description from the description about functional block for a comparison by processing of a low order hierarchy, and performing comparison processing.

[Claim 5] Either [at least] said RTL and netlist comparison means, or said netlist comparison means is formal logic verification equipment according to claim 4 characterized by extracting the description which is equipped with a check list generation means to generate the check list which records a compared block, and is excepted from the object of comparison processing with reference to said check list.

[Claim 6] Said RTL and netlist comparison means perform the comparison about functional block in advance of said netlist comparison means until RTL description and description in

agreement are recognized in a netlist. Said netlist comparison means compares description of a netlist for description of the netlist with which coincidence of logic with RTL description was verified by said RTL and netlist comparison means as the first criteria description. When the inequality during description of a netlist has been recognized by said netlist comparison means, said RTL and netlist comparison means The comparison about functional block is again performed until RTL description and description in agreement are recognized out of the description which is not compared [which is included in a netlist]. Said netlist comparison means Claim 1 characterized by performing the comparison of description of a netlist again by considering description of the new netlist with which coincidence of logic with RTL description was verified as criteria description thru/or formal logic verification equipment of five given in any 1 term.

[Claim 7] Said RTL and netlist comparison means perform the comparison about functional block in advance of said netlist comparison means until RTL description and description in agreement are recognized in a netlist. Said netlist comparison means compares description of a netlist for description of the netlist with which coincidence of logic with RTL description was verified by said RTL and netlist comparison means as the first criteria description. When the inequality during description of a netlist has been recognized by the comparison Claim 1 characterized by resuming the comparison for the description which is not compared [which is included in a netlist considering description of the netlist with which coincidence of logic with RTL description is already verified as the first criteria description] thru/or formal logic verification equipment of five given in any 1 term.

[Claim 8] Said RTL and netlist comparison means generate information, as a result of [1st] expressing the comparison result of the RTL description and the netlist in each comparison point included in the description about functional block. Said netlist comparison means generates information, as a result of [2nd] expressing the comparison result between the netlists in each comparison point included in the description about functional block. And when the inequality of logic has been recognized by said netlist comparison means Claim 1 characterized by having an inequality information generation means to generate the inequality information which expresses the contents of an inequality of description of a netlist, and RTL description based on information said 1st and 2nd results thru/or formal logic verification equipment of seven given in any 1 term.

[Claim 9] It is the formal logic verification approach of comparing RTL description with the

netlist of the gate level obtained by carrying out logic synthesis of said RTL description, and verifying both logic equivalence. When two or more same functional block is contained in a circuit, the RTL description and logic about said functional block until a match is recognized in two or more description about said functional block contained in a netlist The step which compares RTL about the functional block with description of a netlist, After RTL description and the description whose logic corresponds have been recognized in a netlist, description of the netlist as the first criteria description The formal logic verification approach characterized by having a step [mutual / each / of two or more description about said functional block contained in a netlist].

[Claim 10] The formal logic verification approach according to claim 9 characterized by performing the comparison about said functional block gradually towards a high order hierarchy from the lowest hierarchy at either [at least] the step which compares said RTL description and description of said netlist, or the step which compares description of said netlist while having the step which generates the hierarchy of functional block contained in a circuit based on RTL description.

[Claim 11] The formal logic verification approach according to claim 10 characterized by excepting compared description from the description about functional block for a comparison by processing of a low order hierarchy at either [at least] the step which compares said RTL description and description of said netlist, or the step which compares description of said netlist, and performing comparison processing.

[Claim 12] Until RTL description and the description whose logic corresponds are recognized out of the description which is not compared [which is included in a netlist], after the inequality has been recognized between description of a netlist Description of the step which performs the comparison with RTL description and a netlist again, and the new netlist with which coincidence of logic with RTL description was verified by the processing as the first criteria description Claim 9 characterized by performing the step which performs the comparison of description of a netlist again, and ** thru/or the formal logic verification approach of 11 given in any 1 term.

[Claim 13] Claim 9 characterized by resuming the comparison for the description which is not compared [which is included in a netlist] considering description of the netlist with which coincidence of logic with RTL description is already verified as the first criteria description after the inequality has been recognized between description of a netlist thru/or formal logic

verification equipment of 11 given in any 1 term.

[Claim 14] The step which accumulates the comparison result in each comparison point included in the description about functional block, and generates information the 1st result while comparing RTL description with description of a netlist, The step which generates information as a result of [2nd] expressing the comparison result in each comparison point included in the description about functional block, comparing description of a netlist, The step which generates the inequality information which expresses the contents of an inequality of description of a netlist, and RTL description based on information said 1st and 2nd results when the inequality of logic has been recognized between description of a netlist, Claim 9 characterized by preparation ****** thru/or the formal logic verification approach of 13 given in any 1 term.

れば、回路の形式的論理検証を効率的に行うことができる。

【００６８】請求項５記載の発明によれば、チェックリストを参照することで、比較済みの機能ブロックを容易に検知することができる。このため、本発明によれば、比較処理から除外すべきブロックを容易に検知することができる。

【００６９】請求項６または１２記載の発明によれば、ネットリストの記述同士の比較によって論理の不一致が認識された後に、未比較のネットリストの記述を対象として、改めてＲＴＬ記述を基準記述とする形式的論理検証を開始することができる。従って、本発明によれば、ＲＴＬ記述およびネットリストの中に論理不一致の部分が含まれている場合でも、常に全ての機能ブロックについて形式的論理検証を行うことができる。

【００７０】請求項７または１３記載の発明によれば、ネットリストの記述同士の比較によって論理の不一致が認識された後に、既に論理の一致が検証されているネットリストの記述を基準記述として、未比較のネットリストの記述を対象とする形式的論理検証を開始することができる。従って、本発明によれば、常に、全ての機能ブロックについての形式的論理検証を短時間で行うことができる。

【００７１】請求項８または１４記載の発明によれば、ＲＴＬ記述とネットリストの記述との比較に代えて、ネットリストの記述同士の比較を行うこととしているにも関わらず、ＲＴＬ記述とネットリストとの不一致情報を得ることができる。

【図面の簡単な説明】

【図１】　本発明の実施の形態１の形式的論理検証装置＊の処理対象とされる回路のブロック図である。

【図２】　図１に示す回路に関するＲＴＬ記述の１例である。

【図３】　図１に示す回路に関するネットリストの１例である。

【図４】　本発明の実施の形態１において生成されるブロック対応情報の１例である。

【図５】　図１に示す回路に関するＲＴＬ記述の他の例である。

【図６】　本発明の実施の形態２において生成される階層情報の１例である。

【図７】　図１に示す回路に関するネットリストの他の例である。

【図８】　本発明の実施の形態３において用いられるチェックリストの１例である。

【図９】　図８に示すチェックリストに所定のチェックが書き込まれた状態のチェックリストである。

【図１０】　本発明の実施の形態５において不一致情報が生成される手順を説明するための図である。

【図１１】　本発明の実施の形態６の形式的論理検証装置において実行される一連の処理のフローチャートである。

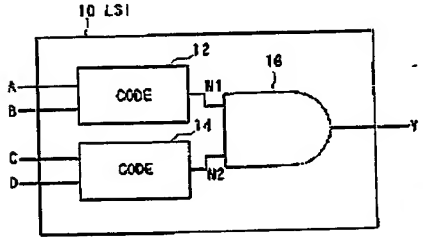
【図１２】　本発明の実施の形態６の形式的論理検証装置における特徴的動作を説明するための図である。

【図１３】　従来の形式的論理検証装置の動作を説明するための図である。

【符号の説明】

１０　ＬＳＩ．　１２、１４　ＣＯＤＥ、　１６　ＡＤＤ。

【図１】



【図２】

```
module LSI ( Y, A, B, C, D ) ;
input A, B, C, D ;
output Y ;

wire N1, N2;

assign Y = N1 & N2 ;

CODE U1 ( N1, A, B ) ;
CODE U2 ( N2, C, D ) ;

endmodule

module CODE ( Y, A, B ) ;
input A, B ;
output Y ;
   (中略 ; RTL記述)
endmodule
```

【図４】

```
CODE -- CODE_1
CODE -- CODE_2
```

【図６】

```
level 0 LSI
level 1 CODE
level 2 ADD
```

【図８】

```
CODE -1      CODE_1 -1
ADD  -1      CODE_2 -1
             ADD_11 -1
             ADD_12 -1
             ADD_21 -1
             ADD_22 -1
```

【図3】

```
module LSI(Y, A, B, C, D);
input A, B, C, D;
output Y;

wire N1, N2;

and (Y, N1, N2);

CODE_1 U1 (N1, A, B);
CODE_2 U2 (N2, C, D);

endmodule

module CODE_1 (Y, A, B);
input A, B;
output Y;
   (中略 : Gate Level Netlist)
endmodule

module CODE_2 (Y, A, B);
input A, B;
output Y;
   (中略 : Gate Level Netlist)
endmodule
```

【図5】

```
module LSI(Y, A, B, C, D);
input A, B, C, D;
output Y;
wire N1, N2;

assign Y = N1 & N2;
CODE U1 (N1, A, B);
CODE U2 (N2, C, D);
endmodule

module CODE (Y, A, B);
input A, B;
output Y;
   (中略 : RTL記述)
endmodule

ADD SU1 (net1, A, X);
ADD SU2 (net2, B, Z);
   (中略 : RTL記述)
endmodule

module ADD (Y, A, B);
input A, B;
output Y;
   (中略 : RTL記述)
endmodule
```

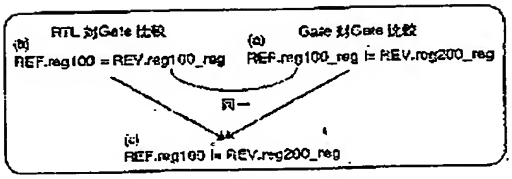
【図7】

```
module LSI(Y, A, B, C, D);
input A, B, C, D;
output Y;
wire N1, N2;

and (Y, N1, N2);
CODE_1 U1 (N1, A, B);
CODE_2 U2 (N2, C, D);
endmodule

module CODE_1 (Y, A, B);
input A, B;
output Y;
ADD_11 SU1 (net1, A, X);
ADD_12 SU2 (net2, B, Z);
   (中略 : Gate Level Netlist)
endmodule

module CODE_2 (Y, A, B);
input A, B;
output Y;
ADD_21 SU1 (net1, A, X);
ADD_22 SU2 (net2, B, Z);
   (中略 : Gate Level Netlist)
endmodule

module ADD_11 (Y, A, B);
   (中略 : Gate Level Netlist)
endmodule

module ADD_12 (Y, A, B);
   (中略 : Gate Level Netlist)
endmodule

module ADD_21 (Y, A, B);
   (中略 : Gate Level Netlist)
endmodule

module ADD_22 (Y, A, B);
   (中略 : Gate Level Netlist)
endmodule
```

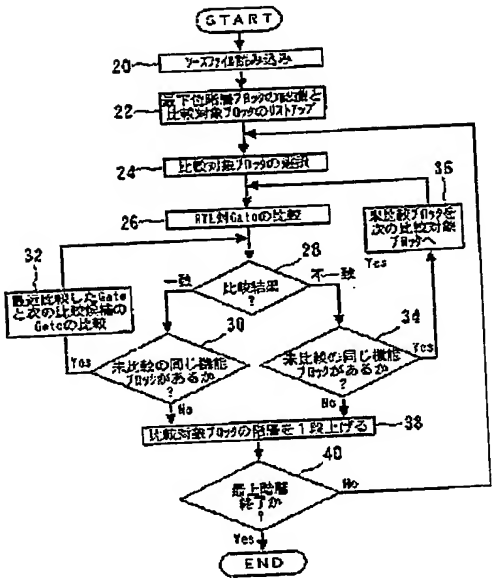
【図9】

CODE -1	CODE_1 -1
ADD 0	CODE_2 -1
	ADD_11 0
	ADD_12 0
	ADD_21 0
	ADD_22 0

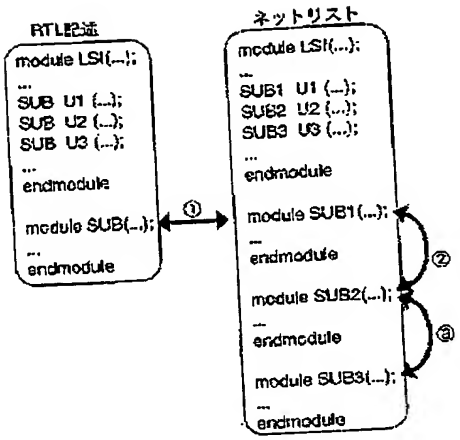
【図10】



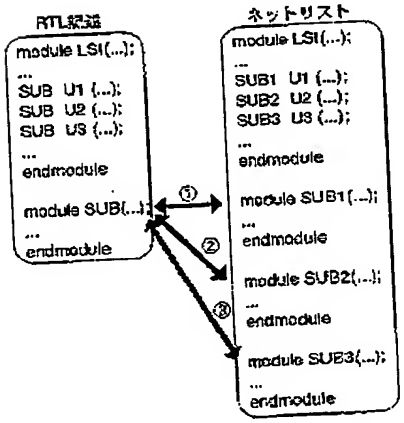
【図１１】


START
20
ソースファイル読み込み
22
下位階層ブロックの記憶と比較対象ブロックのリストアップ
24
比較対象ブロックの選択
35
RTL対Gateの比較
次の比較対象ブロックへ
Yes
32
28
比較結果?
一致
不一致
直近比較したGateと次の比較対象のGateの比較
30
34
未比較の同じ機能ブロックがあるか?
Yes
No
比較対象ブロックの階層を１段上げる
38
40
最上階層終了か?
No
Yes
END

【図１２】


RTL記述
module LSI(...);
...
SUB U1 (...);
SUB U2 (...);
SUB U3 (...);
...
endmodule
module SUB(...);
...
endmodule
①
ネットリスト
module LSI(...);
...
SUB1 U1 (...);
SUB2 U2 (...);
SUB3 U3 (...);
...
endmodule
module SUB1(...);
...
endmodule
②
module SUB2(...);
...
endmodule
③
module SUB3(...);
...
endmodule

【図１３】


RTL記述
module LSI(...);
...
SUB U1 (...);
SUB U2 (...);
SUB U3 (...);
...
endmodule
module SUB(...);
...
endmodule
①
②
③
ネットリスト
module LSI(...);
...
SUB1 U1 (...);
SUB2 U2 (...);
SUB3 U3 (...);
...
endmodule
module SUB1(...);
...
endmodule
module SUB2(...);
...
endmodule
module SUB3(...);
...
endmodule